**AI Based Diabetes Prediction System**

**BATCH MEMBER : K.Aadhi surya -731221104001**

**K.Chandra Prakash -731221104008**

**G.Madhan -731221104022**

**P.Pandiya raj -731221104027**

**V.Bambavasan -731221104007**

**Phase 3: Development Part 1**

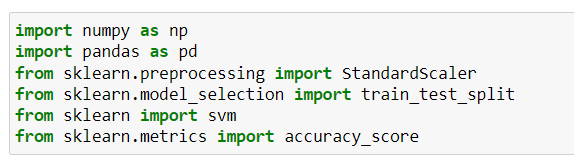
**Loading and preprocessing the dataset**.

**Data Collection:**

* Obtain a diabetes dataset from a reliable source. Common sources include healthcare databases, research institutions, or publicly available datasets.

**Import Libraries:**

* In your Python environment, import the necessary libraries for data manipulation and machine learning. Common libraries include pandas, numpy, and scikit-learn.



**Load the Dataset:**

* Use pandas to load the diabetes dataset into a DataFrame.

**For example:**

import pandas as pd

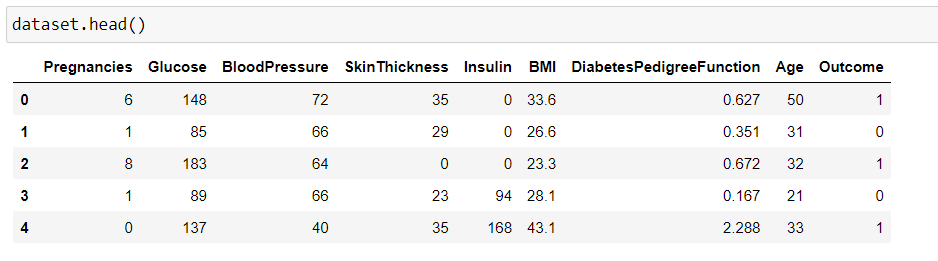
# Load the dataset

data = pd.read\_csv('diabetes\_dataset.csv')

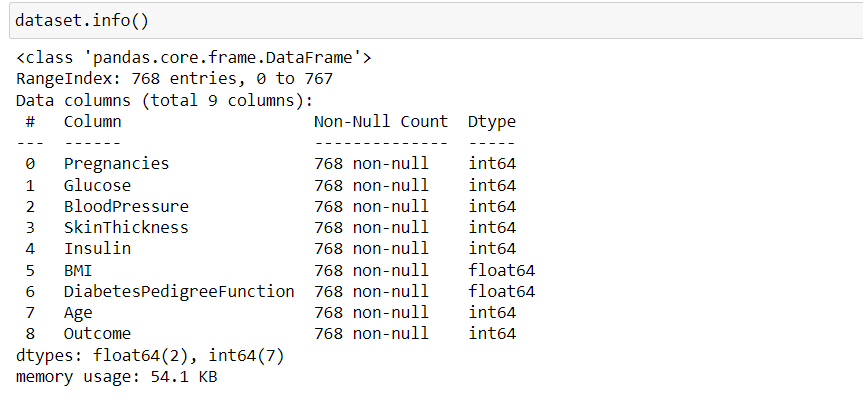


**Data Exploration:**

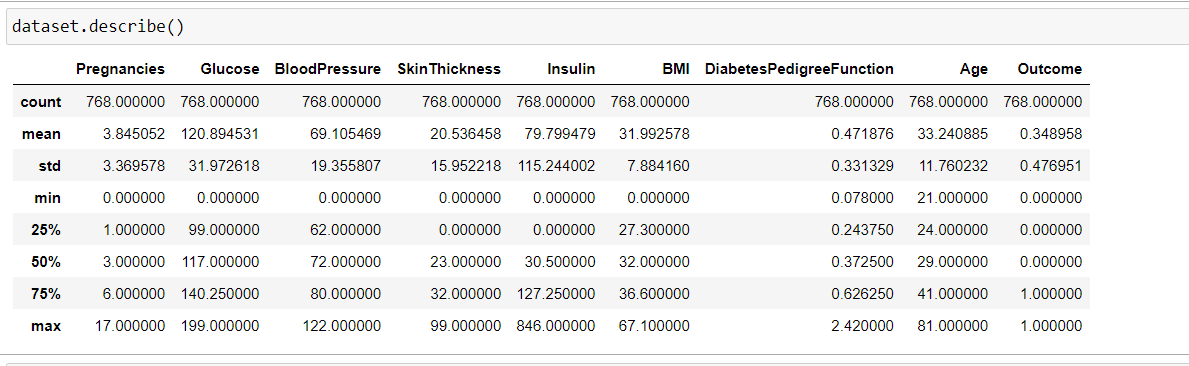
* Explore the dataset to understand its structure and characteristics. This helps you identify any issues that need to be addressed in the preprocessing phase.
* Use **data.head()** to view the first few rows of data.



* Use **data.info()** to check for data types, missing values, and the number of non-null entries.

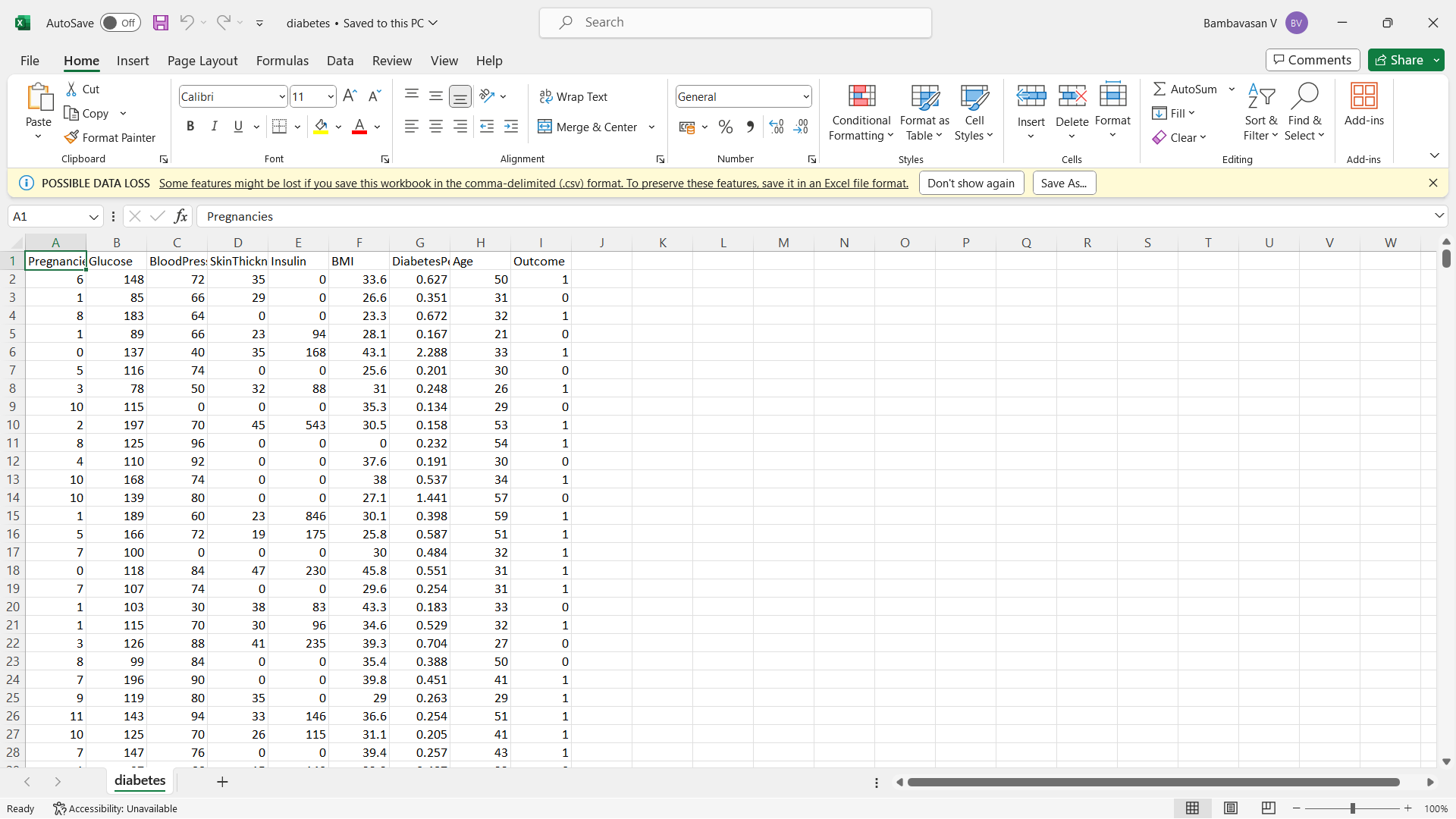


* Use **data.describe()** to get summary statistics of the dataset.



**Data Preprocessing:**

* Prepare the data for modeling by handling missing values, outliers, and encoding categorical variables:
* Handle missing values using techniques such as imputation or removing rows with missing data.
* Check for and address outliers if they exist.
* Encode categorical variables into numerical format (e.g., one-hot encoding for binary categories).

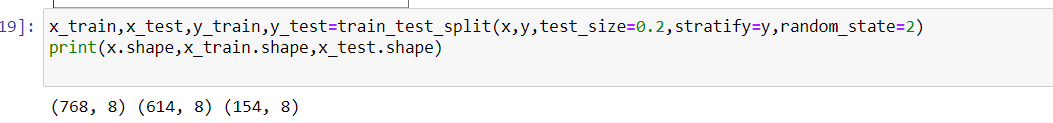


**Feature Selection:**

* Select relevant features for your diabetes prediction model. You can use various techniques for this, such as:
* Correlation Analysis: Identify features that are strongly correlated with the target variable (diabetes status).
* Feature Importance: Use machine learning models like Random Forest or XGBoost to rank features based on their importance.
* Domain Knowledge: Consult experts or medical literature to understand the relevance of different features.

**Data Splitting:**

Split the dataset into a training set and a testing set. A common split ratio is 70-80% for training and 20-30% for testing. This allows you to evaluate your model's performance.



**Save Preprocessed Data**:

Save the preprocessed data into a new file (e.g., a CSV file) so that you can easily access it in the modeling phase.

**Documentation:**

* Document your data preprocessing and feature selection steps. This documentation is essential for reproducibility and future reference.

**Further Data Preparation:**

* Depending on the specific dataset and project requirements, you might need to perform additional steps like feature scaling, dimensionality reduction, or creating new features based on domain knowledge.